

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	67	shar\$3 near8 web\$1 near8 session\$1	US-PGPUB; USPAT	OR	ON	2004/06/28 16:18
S2	26	(shar\$3 near8 web\$1 near8 session\$1) and @ad<"20010217"	US-PGPUB; USPAT	OR	ON	2004/11/17 17:08
S3	27	(shar\$3 near8 web\$1 near8 session\$1) and @ad<"20010217"	US-PGPUB; USPAT	OR	ON	2004/11/17 17:08
S4	5	S3 and prox\$3	US-PGPUB; USPAT	OR	ON	2004/11/17 17:08

US-PAT-NO: **6748420**

DOCUMENT-IDENTIFIER: US 6748420 B1

TITLE: Methods and apparatus for providing shared access to an application

----- KWIC -----

Abstract Text - ABTX (1):

The system of the invention provides a collaboration adapter which can be integrated into a web server or a web/application server to allow multiple computer user participants to access a single shared session to an application in a collaborative manner. The system of the invention allows a participant to create a shared session and to associate a participant identity to the shared session as the shared session owner. From thereon, participants can join the **shared session and submit requests for application response information to the web** server containing the collaboration adapter. In cases where the application response information must be obtained from the application, the invention substitutes participant session identification information in an original participant request with shared session identification information to create an altered request and then forwards the altered request to the application.

US Patent No. - PN (1):  
**6748420**

Application Filing Date - AD (1):  
**20001011**

Brief Summary Text - BSTX (2):

The present invention generally relates to systems for providing shared access to a software application, and more particularly, to systems and techniques which provide concurrent access by multiple participants to a single **shared session of a software application served by a web** server.

Brief Summary Text - BSTX (21):

The present invention is preferably embodied in part as a collaboration adapter that may be incorporated into a web server. In this capacity, the system of the invention is able to create a collaborative shared session that allows two or more independent **sessions (e.g., two different computer user HTTP sessions) to share a single shared application session existing in the web** or application server. Generally, the system of the invention regulates access to the single shared session by multiple session participants by including a copy service that stores or caches application response information (e.g., web pages containing transactional information) received from the application (e.g., in response to an initial request from a participant for the information) such that the application response information can be provided repeatedly to each requesting participant to the collaborative shared session in the event that those participants request the same information.

Brief Summary Text - BSTX (23):

The system of the invention provides the **shared session between the web server and the application by creating shared** session identification information which can include, for example, shared session cookies as well as shared session participant identification information for each participant computer user who accesses the application via the shared session. That is,

the invention provides the application with the appearance of a single user session (i.e., the shared session) that is controlling the application.

Brief Summary Text - BSTX (40):

In yet other embodiments, the steps of forwarding the application response information include the step of determining an identity of a participant requesting the application response information and altering content of the application response information based on the identity of the participant requesting the application response information and then forwarding the application response information to the participant requesting the application response information. In this manner, the invention can determine who is requesting a web page, for example, and can alter the contents of the web page within the shared session for that requesting participant, while other participants might receive differently altered pages. The identity (e.g., a cookie) of a participant can be used to determine who the participant is and thus can be used to select from various content alterations.

Brief Summary Text - BSTX (47):

In still other embodiments, the shared session identification information is distributed and accessible by multiple web servers. This allows an e-commerce web site that employs the system of the invention to be distributed, for example, behind a load balancer.

Brief Summary Text - BSTX (48):

Other embodiments of the invention include computer system arrangements providing participant access to an application. According to one such embodiment, the computer system arrangement includes an input/output interface, at least one memory system, and a processor coupled to the input/output interface and the memory system. The memory system (there may be more than one) is encoded with an application and a web server process including a collaboration adapter that, when performed on the processor, causes the computer system arrangement to perform the operations of receiving, over the input/output interface, a first request from a first participant via a first participant session to access the application via a shared session between the web server and the application. When further performed, the collaboration adapter process substitutes, in the memory system, participant session identification information identifying the first participant session within the first request with shared session identification information identifying the shared session to create an altered request and forwards the altered request to the application such that the application operates to produce application response information in accordance with shared session identification information contained in the altered request, independently of participant session identification information contained in the first request.

Drawing Description Text - DRTX (9):

FIG. 7 illustrates an alternative architecture of an e-commerce web site configured in accordance with one embodiment of the invention that uses a collaboration framework to indicate shared session events to participants and a web server operating a shared session.

Detailed Description Text - DETX (5):

In FIG. 1, assume that user browsers 10, 20, 22 and 24 independently access the web server 30 initially, but that they collectively at some point need to view the same dynamic, personalized, secure and transactional-based web pages in order to work together synchronously. Using the system of the invention, along with any HTTP request, participant 10 (by way of example only) can initiate collaboration by requesting the creation of a named HTTP shared session 16 to be created to the application server 50 served by the web server 30.

Detailed Description Text - DETX (7):

In step 391, if not already present, the collaboration adapter 200 creates an HTTP **shared session 16 between the web** server 30 and the application server 50, and adds participant 10 as a participant to this shared session. The creation process for a shared session and the addition of participant to a shared session are discussed in more detail with respect to FIGS. 4 and 5.

Detailed Description Text - DETX (12):

In step 394, the collaboration adapter 200 then returns control to the web server 30 and the **web server 30 forwards the request (the altered request) containing the shared session** identification information to the application server 50 which operates in accordance with participant identification information defined within the shared session identification information, irrespective of any participant identity specified in the original request (received in step 390). At this point, the application server 50 is unaware of any differences in handling the HTTP request using a participant identity associated with the **shared session 16 and simply executes program logic and returns, for example, a web** page (not specifically shown) containing application response information to the web server 30.

Detailed Description Text - DETX (17):

More specifically, the participant 20 via user session 13 forwards an HTTP join-shared-**session request (not specifically shown) containing the name of the shared session 16 to the web** server 30. The web server 30, in step 390, receives and forwards this subsequent or second request to the collaboration adapter 200. Since the shared session 16 already exists, the collaboration adapter 200 does not create a new shared session in step 391, but rather finds the shared session 16 by name and since this is a new participant to the shared session 16, associates the new HTTP participant session 13 for participant 20 to the shared session 16 in step 391. Just as for participant 10, the collaboration adapter 200 in step 391 also generates a shared session-participant ID for participant 20.

Detailed Description Text - DETX (18):

The collaboration adapter 200 then determines in step 392 if the requested application response information is present in the collaboration adapter 200. Assume for this example that the requested information is present and so processing proceeds to step 397 where the collaboration adapter 200 forwards a copy of previously obtained application response information obtained from the application server 50 for the original request from participant 10) cached for the **shared session 16 to the web server 30 which returns this page to the new shared** session participant 20. Along with the application response information, the collaboration adapter (via the **web server 30**) includes the **unique shared session** ID for shared session 16 as well as the shared session participant ID generated for the new participant 20, uniquely identifying this participant as being part of the shared session 16. Since participant 20 is not the owner of the shared session 16, any previous application specific cookies generated by the application server 50 within the resulting page are not forwarded to the browser of participant 20.

Detailed Description Text - DETX (22):

The collaboration adapter 200 receives the request information (step 393) and finds the identified shared session 16 (e.g., by its name, cookie, or other identity), authenticates the HTTP session 13 as being a valid participant in the shared session 16, and validates the request. In this embodiment, the collaboration adapter ignores any application cookies within the original request from participant 20. Instead, the collaboration adapter 200 retrieves shared session identity information (not shown in this Figure, but maintained

within the collaboration adapter 200) associated with the owner (e.g., participant 10) of the shared session 16 and substitutes or overwrites (step 393) the original HTTP session state of the request with cached session state (shared session identification information) associated with the shared session 16 (i.e., session state associated with owner participant 10 and shared session 16). Essentially, the collaboration adapter 200 causes the web server 30 to construct an altered request under the identity of the shared session 16, instead of the participant session 13. It is to be understood that the specific information passed between requests received from participants through the collaboration adapter to the application may be configurable such that unique participant identity and/or other information (cookie) may or may not be provided to the application.

Detailed Description Text - DETX (26):

Continuing with the example, next, participant 10 needs to get the shared result page that the application server 50 generates based on the former request for the next page that has already been processed by the web server 30 for participant 20. Participant 10 becomes aware of the new or next page using mechanisms discussed in detail later. Once the participant 10 is aware of the new page in the shared session 16, the participant 10 requests the current page via session 12 to the web server 30 (step 390). Because the request is accompanied by the shared session and shared session-participant cookies, the web server 30 forwards the request to the collaboration adapter 200 and step 391 can be skipped completely by the collaboration adapter 200 (since the shared session exists and participant 10 is already a member).

Detailed Description Text - DETX (33):

The collaboration adapter 200 of the invention handles, among other tasks, the management of the shared session 16 between the web server 30 and the application server 50. In certain embodiments, the application server 50 is completely unaware that there are multiple participants to the shared session and is thus only aware of a single user identity controlling the shared session 16. This allows collaboration to occur without any changes being required in application logic within the application server 50.

Detailed Description Text - DETX (34):

The web server 30 configured according to the invention serves copies of pages containing the application response information that, in most embodiments, are identical to those produced by an application server 50 operating in a conventional non-collaborative environment. That is, copies of web pages within the shared session 16 are served to shared session participants 10 through 24 and these pages include live hypertext links and active buttons directing all of the browsers of the shared session participants back to be original web server 30, rather than to a proxy server. Since the application server 50 generally is aware of only one user session guiding application transaction processing (i.e. the shared session 16), the application server 50 can operate in a completely conventional manner without any modifications.

Detailed Description Text - DETX (44):

According to this embodiment of the invention, in step 400 in FIG. 4, the participant 10 (via that participant's web browser, for example) sends an HTTP request 100 to the web server 30 that requests the creation of a named HTTP shared session which all participants will use to access to the application server 50. The command to request creation of a shared session may be embedded along with any HTTP request. For the request 100 containing the "create shared session command", in step 401, the HTTP handler interceptor 32 intercepts the request 100 and passes the request information 210 to the collaboration adapter 200.

Detailed Description Text - DETX (47):

In step 402, the collaboration adapter 200 also creates participant session identification information 206-1 within the shared session state 205 for participant 10. When a participant (e.g., 10, 20) creates and/or initially joins an existing shared session, the collaboration adapter 200 creates a set of participant session identification information 206 for that participant. A set of participant session identification information 206 contains a link, reference, pointer or other association to the participant's session state 34 (e.g., 34-1) associated with the participant (e.g., 10) in the **shared session, as well as any other identification information required for the web server 30** to format or identify responses (e.g., 105, 115, as we explained) provided from the web server 30 to a participant browser as being associated with the participant who originated the request for such a response. In other words, the collaboration adapter 200 can use a set of participant session identification information 206 to properly format application response information provided from the application server 50 via the shared session so as to be properly accepted by a participant browser. Participant session states 34-1 through 34-N may generally be considered information related to individual participant user sessions 12 through 15 in FIG. 1.

Detailed Description Text - DETX (58):

An example of a copy server system that can maintain copies of **web pages on behalf of multiple participants to a shared collaboration session** is described in detail in co-pending U.S. patent application Ser. No. 09/347,870, filed Jul. 6, 1999, entitled "Copy Server for Collaboration and Electronic Commerce," which is assigned to the same assignee as the present invention. The entire disclosure, teachings and contents of reference U.S. patent application Ser. No. 09/347,870 are hereby incorporated by reference herein in their entirety. Such a copy server system can provide the functionality of this invention that serves copied pages of application response information to participants of the shared session as explained therein. This reference patent application also provides information on maintaining cookie state between participant browsers and the copy server.

Detailed Description Text - DETX (59):

Now that the initial request 100 has been processed by the application server 50, the **web server 30 is able to provide the results to participant(s) to the shared session**. In step 408, the collaboration adapter 200 causes (via 214) the HTTP handler interceptor 32 to return a copy 214 of the application response information 125 as a resulting web page 105 to the participant 10 that originated the HTTP request 100. Along with the resulting web page 105 containing a copy of the application response information 125, in step 408, the collaboration adapter 200 also returns any application cookies provided by the application server 50 within the application response information 125, since the requesting participant 10 happens to be the owner/creator of the shared session. The collaboration adapter 200 also return a copy of the unique shared session identity/ID and a copy of the shared session participant identity/ID, which were each generated by the collaboration adapter in step 402 during creation of the shared session and addition of participant 10 to the shared session.

Detailed Description Text - DETX (61):

At this point in processing, the participant 10 has established a one-participant **shared session in the web server 30** to interact with the application server 50. The participant 10 may be considered, for purposes of this example, to be the owner or controller of the shared session. The **web server 30 can subsequently add more participants to the shared session**.

Detailed Description Text - DETX (65):

In step 421, the participant 20 (via his or her user browser) establishes a user session (e.g., an HTTP session) with the web server 30 and sends a request 110 to join the shared session, for example, by specifying the name of the shared session within a "join shared session" command contained in an HTTP request 110. As explained above with respect to participant 10, creation of a participant session with the web server 30 involves the HTTP handler interceptor 32 creating another HTTP session state (e.g., 34-2, not specifically numbered) for the HTTP user session (e.g., 13 in FIG. 1) associated with participant 20.

Detailed Description Text - DETX (72):

As explained above, steps 403 through 407 generally cause the web server 30 to obtain the requested page on behalf of the shared session. Since the request 120 is made by the web server 30 on behalf of the shared session, the collaboration adapter 200 substitutes participant identification information 206-2 in the request 110 (participant 20's request) within the HTTP session state 34-2 associated with participant 20 (step 403) with shared session identification information 207 (request, cookie and identity information associated with owner participant 10), such that the original request 110 is converted into an altered request 120 that appears, from the perspective of the application server 50, to be originating on behalf of the identity associated with the shared session. After step 426 initiates and completes processing of steps 403 through 407, the collaboration adapter 200 returns control to step 428 which is explained below.

Detailed Description Text - DETX (74):

Finally, in step 428, the HTTP handler interceptor 32 returns the resulting web page 115 to the participant 20 (i.e., to the participant 20 associated with the session 34-2 in which the request 110 resides). The resulting web page 115 includes the shared session cookie from the shared session identification information 207 identifying the shared session state 205 and the shared session participant cookie from the participant identification information 206-2 (for participant 20). However, since participant 20 is not the owner of the shared session, the collaboration adapter 200 excludes from the resulting web page 115 any application-specific cookies contained within the application response information 208. As such, after the processing of step 428 is complete, the browser associated with participant 20 will contain the unique shared session participant identification cookie which can uniquely identify the participant 20, as well as a cookie identifying the shared session to which that participant is a member, but will not contain any application specific cookies.

Detailed Description Text - DETX (79):

Upon creation of a shared session, the collaboration adapter 200 establishes the shared session identification information 207 containing a unique shared session identity 207-B on behalf of the participant claiming ownership of the shared session, which initially is the participant who creates the shared session. At this point, the collaboration adapter 200 causes the shared session identity 207-B to be exchanged between the Web server 30 and the application server 50 when handling all participant requests (e.g., 100, 110). The collaboration adapter 200 does not exchange the shared session identity 207-B with the participant browsers 10, 20. The collaboration adapter 200 also creates a unique shared session cookie 207-A that uniquely identifies the shared session.

Detailed Description Text - DETX (82):

Generally, the web server 30 serves pages of application response information 208 to shared session participant absent the application specific cookies 54-A. However, the collaboration adapter 200 does allow the

application-specific cookies 54-A to be returned by the **Web server 30 in pages served to the participant browser who is the owner of the shared session** (e.g., participant 10 in the former examples). In other words, none of the participants to the shared session other than the shared session owner receive application specific cookies 54-A while under control of the shared session. This allows each participant to preserve their own identity for subsequent self-service interaction with the Web and application servers 30, 50. In other words, if owner participant of a shared session interacts with the application server 50 a later time, for instance after a shared session has ended, the application-specific cookies 54-A will be current and up-to-date with respect to the participant. However, other participants (e.g., 20) to the shared session may be unknown to exist from the perspective of the application server 50 since they will not contain copies of the application-specific cookies 54-A.

Detailed Description Text - DETX (94):

The collaboration adapter 200 can control participant access to a shared session via the use of an access control list (ACL) 301. The participant owner of a shared session is generally responsible of the creation of the ACL. Once a participant owner of a shared session creates an ACL ownership session, the collaboration adept or 200 treats the shared session as a restricted shared session. An ACL can identify each participant allowed to join and interact with the **shared session and can annual rate operations are excepted from each participant by the Web** server 30. For example, an ACL may specify that a particular participant be allowed to read a page of application response information 208-1 but may not be allowed to submit requests for other pages based on that page 208-1.

Detailed Description Text - DETX (98):

Generally, a **web server 30 configured in accordance with invention provides shared session** control interfaces 302, 304 and 218 to support third party control and/or **shared session event notification related to shared sessions within the web** server 30. In this example, the media blender/collaboration framework 300 enables the participants 10, 20 to obtain the name of the collaborative shared session(s) 205 via communications channels 310, 320 since the framework 300 interfaces 218 with the collaboration adapter 200 and is aware of the existence of shared session state(s) 205 (one for each shared session). In other words, the collaboration adapter (also referred to as a controller in the patent application having Ser. No. 8/852,764) can indicate which shared session exist to the framework 300 and the framework 300 can relay this information to the participants 10, 20 or any other participants of one or more other shared sessions.

Detailed Description Text - DETX (99):

The collaboration adapter 200 can also make the framework 300 aware of any new pages of application response information 208 within a shared session which the collaboration adapter 200 caches when such pages are returned from the application server 50 during the shared session. As such, the framework 300 can manage the communication channels 310,320 to notify the participants 10, 20 of new or existing sessions, and when a participant joins one or more of such sessions, the framework 300 indicates to those participants in a session when a new page of application response information 208 is available for retrieval from the web server 30. The participants 10, 20 can then request 110, 115 the next page of application response information 208 from the **web server 30 and the requests 110,115 can be accompanied by the shared session** and shared session participant identification information.

Detailed Description Text - DETX (100):

In an alternative embodiment, a privileged user can control a **shared session once the privileged user is authenticated to the web** server 30. The media



blender/collaboration framework 300 is an example of a privileged user. Within the collaboration adapter 200, the session tracker 302 can send and receive external event information 301 to control shared session events, operation and progression. The framework 300 includes shared session management services (not specifically shown) that can use the HTTP communications channel API 302, 304 and 218 in conjunction with the session tracker 302 to create and terminate shared sessions, add or remove participants from shared sessions, define shared session owners, create and modify shared session ACL's, detect changes in shared session page content and generally control the collaboration adapter 200.

Detailed Description Text - DETX (103):

In an alternative embodiment, a privileged user such as the media Bender/collaboration framework 300 can terminate in a shared session based on such factors as inactivity, in each or some other external event. If a shared session is terminated in this manner, the collaboration adapter 200 is notified of the termination event via the session tracker 302 and each shared session participant 10, 20 is dropped from the shared session and the shared session state 205 is cleared from the cache memory. In this manner, the lifetime of a shared session participant with respect to the shared session is controlled by the lifetime of the participants HTTP session with the web server 30.

Detailed Description Text - DETX (104):

In addition to complete shared session termination, any shared session participant can voluntarily terminate his or participation in a shared session by either terminating that participants HTTP session with the web server 30, or by specifying an exit-shared-session-participant command within an HTTP request associated with the shared session. When a participant exits a shared session, the participants original HTTP session with the Web server 30 can remain intact and the collaboration adapter 200 will discontinue altering the HTTP session state 34 associated with that participants session with the Web server 30. That is, the collaboration adapter 200 will no longer alter requests 100, 110 made by a participant who exits a shared session.

Detailed Description Text - DETX (107):

In order to support a shared session across multiple web servers 30 in this manner, the system of the invention provides a distributed shared session state 205 that each collaboration adapter 200-1 through 200-P can access via communication channels 146,148. The system of the invention can distribute the distributed shared session state 205 by replication methods or by providing a centralized shared sessions state server (not shown) that is accessible by each collaboration adapter 200. In this manner, the collaboration adapters 200 within each web server 30 can access all participant information related to the shared session, no matter which web server 30 receives or handles a request for a participant operation related to the shared session.

Detailed Description Text - DETX (112):

FIG. 10 illustrates an example configuration of an embodiment of the invention in which multiple participants may share pages of application response information 208 related to a shared session concurrently wherein such pages 208 may be presented as different views to different participants. To do so, the system of the invention can identify various roles associated with different participants to a shared session. For example, one participant might be an agent or representative of the company or other institution providing the e-commerce web site 25, while other participants to the shared session may be customers requiring assistance while interacting with the application 58.

Detailed Description Text - DETX (115):

The present invention provides a system which allows a web server to provide

**shared access by multiple participants to a single session** of interaction with an application. This allows a single application session to be viewed and controlled by multiple participant computer users over a network such as the Internet. The present invention is particularly useful in the e-commerce web site platforms which benefit from collaborative sessions in which multiple participant computer users can interact with a single communications session directed to an application server.

Claims Text - CLTX (7):

7. The method of claim 1, wherein the first participant uses a web browser to generate the first request and the application is run on a web-based server device that generates and transmits the response information depending on the first request, the application receiving additional requests from other participants of the **shared session, the web**-based server supporting concurrent access by the other participants to response information generated on behalf of the first participant.

Claims Text - CLTX (27):

27. The method of claim 16, wherein a first participant uses a **web browser to generate a first request to create a shared session, and the application is run on a web**-based server device that generates and transmits the application response information depending on the first request, the application receiving additional requests from other participants of the **shared session, the web**-based server supporting concurrent access by the other participants to application response information generated on behalf of the first participant.

Claims Text - CLTX (31):

31. A computer system providing participant access to an application, the computer system comprising: an input/output interface at least one memory system; and a processor coupled to the input/output interface and the at least one memory system; the at least one memory system encoded with an application and a server process including a collaboration adapter that, when performed on the processor, causes the computer system to perform the operations of: receiving, over the input/output interface, a first request from a first participant via a first participant **session to access the application via a shared session between the web server and the application, the shared** session allowing access to the application by at least one participant; substituting, in the at least one memory system, participant session identification information identifying the first participant session within the first request with shared session identification information identifying the shared session to create an altered request; forwarding the altered request to the application such that the application operates to produce application response information in accordance with shared session identification information contained in the altered request, independently of participant session identification information contained in the first request; and determining if application response information appropriate to the request is present in a cache of application response information, and if so, forwarding the application response information appropriate to the request from the cache to the participant via the participant session such that the request does not have to be processed by the application via the shared session, and if not, performing the steps of substituting and forwarding to obtain application response information appropriate to the request.

Claims Text - CLTX (40):

40. A computer system providing participant access to an application, the computer system comprising: an input/output interface at least one memory system; and a processor coupled to the input/output interface and the at least one memory system; the at least one memory system encoded with an application and a server process including a collaboration adapter that, when performed on

the processor, causes the computer system to perform the operations of: receiving, over the input/output interface, a first request from a first participant via a first participant session to access the application via a shared session between the web server and the application, the shared session allowing access to the application by at least one participant; substituting, in the at least one memory system, participant session identification information identifying the first participant session within the first request with shared session identification information identifying the shared session to create an altered request; and forwarding the altered request to the application such that the application operates to produce application response information in accordance with shared session identification information contained in the altered request, independently of participant session identification information contained in the first request; and determining if the first participant is an owner of the shared session, and if so, including any application specific state information within the application response information forwarded to the first participant, and if not, excluding any application specific state information within the application response information forwarded to the first participant.